\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Experiment No.: 1

# Name of Expt.: Write a program to demonstrate status of key on an Applet window such as KeyPressed, KeyReleased, KeyUp, KeyDown.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Program –**

**KeyStatus.java -**

import java.awt.\*;

import java.applet.\*;

import java.awt.event.\*;

public class KeyStatus extends Applet implements KeyListener

{

String msg = "";

public void init()

{

addKeyListener(this);

}

public void keyReleased(KeyEvent k)

{

showStatus("Key Released");

repaint();

}

public void keyTyped(KeyEvent k)

{

showStatus("Key Typed");

repaint();

}

public void keyPressed(KeyEvent k)

{

showStatus("Key Pressed");

repaint();

}

public void paint(Graphics g)

{

g.drawString(msg, 10, 10);

}

}

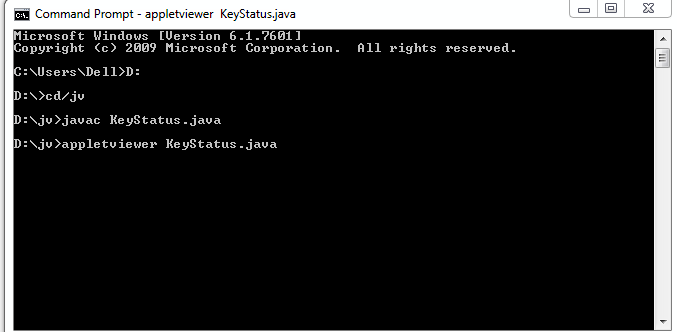
/\*

<applet code="KeyStatus.class" height="400" width="400">

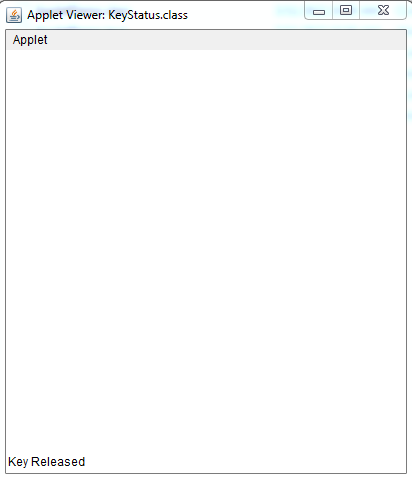
</applet>

\*/

**Command –** Run in cmd promt.



**Output –**



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Experiment No.: 2

# Name of Expt.: Write a program to create a frame using AWT. Implement mouseClicked, mouseEntered() and mouseExited() events. Frame should become visible when the mouse enters it. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Program –

# Mouse.java -

# import java.awt.\*;

# import java.awt.event.\*;

# public class Mouse extends Frame implements MouseListener{

# Label l;

# Mouse(){

# addMouseListener(this);

# 

# l=new Label();

# l.setBounds(20,50,100,20);

# add(l);

# setSize(300,300);

# setLayout(null);

# setVisible(true);

# }

# public void mouseClicked(MouseEvent e) {

# l.setText("Mouse Clicked");

# }

# public void mouseEntered(MouseEvent e) {

# l.setText("Mouse Entered");

# }

# public void mouseExited(MouseEvent e) {

# l.setText("Mouse Exited");

# }

# public void mousePressed(MouseEvent e) {

# l.setText("Mouse Pressed");

# }

# public void mouseReleased(MouseEvent e) {

# l.setText("Mouse Released");

# }

# public static void main(String[] args) {

# new Mouse();

# }

# }

# Command – Run in cmd prompt.

# D:\2 NSB WORK\NSB ICOER\ICOER 21-22\AJP\Lab\All programs\program 2\command 2.PNG

# Output –

# D:\2 NSB WORK\NSB ICOER\ICOER 21-22\AJP\Lab\All programs\program 2\output 2.PNG

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Experiment No.: 3

# Name of Expt.: Develop a GUI which accepts the information regarding the marks for all the subjects of a student in the examination. Display the result for a student in a separate window. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Program –**

**Fee.java -**

**package** demo;

**import** javax.swing.\*;

**import** java.awt.\*;

**import** java.awt.Image;

**import** java.awt.event.\*;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** java.awt.print.\*;

//import javafx.print.Printer;

**import** java.io.\*;

**import** java.io.IOException;

// Creating the fee class

**public** **class** Fee **extends** Frame {

JLabel l1, l2, l3, l4,

l5, l6, l7, l8,

l9, l10, l12, l13,

l14, l11, l15;

JTextField tf1, tf2, tf3,

tf4, tf5, tf6,

tf7, tf8, tf9,

tf10;

JTextArea area2, area1;

JRadioButton rb1, rb2, rb3,

rb4, rb5, rb6,

rb7;

JFileChooser f1;

// Default constructor to

// initialize the parameters

Fee()

{

l1 = **new** JLabel("Student Report");

l1.setBounds(550, 100, 250, 20);

l2 = **new** JLabel(

"Name of the Student:");

l2.setBounds(50, 150, 250, 20);

tf1 = **new** JTextField();

tf1.setBounds(250, 150, 250, 20);

l3 = **new** JLabel(

"Subject 1:");

l3.setBounds(50, 200, 250, 20);

tf2 = **new** JTextField();

tf2.setBounds(250, 200, 250, 20);

l4 = **new** JLabel("Subject 2:");

l4.setBounds(50, 250, 250, 20);

tf3 = **new** JTextField();

tf3.setBounds(250, 250, 250, 20);

l5 = **new** JLabel("Subject 3:");

l5.setBounds(50, 300, 250, 20);

tf4 = **new** JTextField();

tf4.setBounds(250, 300, 250, 20);

l6 = **new** JLabel("Contact Number:");

l6.setBounds(50, 350, 250, 20);

tf5 = **new** JTextField();

tf5.setBounds(250, 350, 250, 20);

l7 = **new** JLabel("Address:");

l7.setBounds(50, 400, 250, 20);

area1 = **new** JTextArea();

area1.setBounds(250, 400, 250, 90);

l9 = **new** JLabel("Gender:");

l9.setBounds(50, 500, 250, 20);

JRadioButton r5

= **new** JRadioButton(" Male");

JRadioButton r6

= **new** JRadioButton(" Female");

r5.setBounds(250, 500, 100, 30);

r6.setBounds(350, 500, 100, 30);

ButtonGroup bg = **new** ButtonGroup();

bg.add(r5);

bg.add(r6);

l10 = **new** JLabel("Nationality:");

l10.setBounds(50, 550, 250, 20);

tf6 = **new** JTextField();

tf6.setBounds(250, 550, 250, 20);

l11 = **new** JLabel(

"Year of passing 10th");

l11.setBounds(50, 600, 250, 20);

String language[]

= { "2016", "2015", "2014" };

**final** JComboBox cb1

= **new** JComboBox(language);

cb1.setBounds(250, 600, 90, 20);

l12 = **new** JLabel(

"Year of passing 12th");

l12.setBounds(50, 650, 250, 20);

String languagess[]

= { "2019", "2018", "2017" };

l13 = **new** JLabel(

"Pin Code:");

l13.setBounds(50, 700, 250, 20);

tf7 = **new** JTextField();

tf7.setBounds(250, 700, 250, 20);

l14 = **new** JLabel("Percentage in 12th:");

l14.setBounds(50, 750, 250, 20);

tf8 = **new** JTextField();

tf8.setBounds(250, 750, 250, 20);

ImageIcon i2 = **new** ImageIcon("2.png");

JLabel l15

= **new** JLabel("", i2, JLabel.***CENTER***);

l15.setBounds(900, 50, 600, 200);

**final** JComboBox cb2

= **new** JComboBox(languagess);

cb2.setBounds(250, 650, 90, 20);

l8 = **new** JLabel(

"Branch:");

l8.setBounds(800, 150, 250, 20);

rb1 = **new** JRadioButton("ICOER");

rb1.setBounds(550, 150, 100, 30);

rb2 = **new** JRadioButton("BSIOTR");

rb2.setBounds(650, 150, 100, 30);

ButtonGroup bg1 = **new** ButtonGroup();

bg1.add(rb1);

bg1.add(rb2);

rb3 = **new** JRadioButton("Good");

rb3.setBounds(550, 200, 100, 30);

rb4 = **new** JRadioButton("Improvement");

rb4.setBounds(650, 200, 120, 30);

ButtonGroup bg2 = **new** ButtonGroup();

bg2.add(rb3);

bg2.add(rb4);

String languages[]

= { "COMP", "ENTC", "EE",

"CIVIL", "MECH" };

**final** JComboBox cb

= **new** JComboBox(languages);

cb.setBounds(800, 200, 90, 20);

**final** JLabel label

= **new** JLabel();

label.setBounds(600, 430, 500, 30);

JButton b = **new** JButton("Show");

b.setBounds(1000, 300, 80, 30);

**final** DefaultListModel<String> li1

= **new** DefaultListModel<>();

li1.addElement("COMP(2, 50, 000)");

li1.addElement("ENTC(2, 50, 000)");

li1.addElement("EE(2, 50, 000)");

li1.addElement("MECH(2, 50, 000)");

li1.addElement("CIVIL(2, 50, 000)");

**final** JList<String> list1

= **new** JList<>(li1);

list1.setBounds(600, 300, 125, 125);

DefaultListModel<String> li2

= **new** DefaultListModel<>();

li2.addElement(

"2 SHARE(1, 50, 000)");

li2.addElement(

"3 SHARE(1, 40, 000)");

li2.addElement(

"5 SHARE(1, 20, 000)");

li2.addElement(

"8 SHARE(1, 10, 000)");

// li2.addElement(

// "bus(40, 000)");

**final** JList<String> list2

= **new** JList<>(li2);

list2.setBounds(

800, 300, 125, 125);

JButton Receipt

= **new** JButton("Generate Receipt");

Receipt.setBounds(600, 490, 150, 30);

JButton b2 = **new** JButton("Reset");

b2.setBounds(750, 490, 150, 30);

JButton Print = **new** JButton("Print");

Print.setBounds(900, 490, 150, 30);

area2 = **new** JTextArea();

area2.setBounds(600, 540, 450, 240);

add(l1);

add(l2);

add(l3);

add(l4);

add(l5);

add(l6);

add(l7);

add(l8);

add(l9);

add(l10);

add(l11);

add(l12);

add(l13);

add(l14);

add(tf1);

add(tf2);

add(tf3);

add(tf4);

add(tf5);

add(tf6);

add(tf7);

add(tf8);

add(area1);

add(area2);

add(l15);

add(rb1);

add(rb2);

add(rb3);

add(rb4);

add(r5);

add(r6);

add(cb);

add(cb1);

add(cb2);

add(list1);

add(list2);

add(b);

add(label);

add(Receipt);

add(b2);

add(Print);

b.addActionListener(**new** ActionListener() {

// Method to display the data

// entered in the text fields

**public** **void** actionPerformed(ActionEvent e)

{

String data = "";

**if** (list1.getSelectedIndex() != -1) {

data = "You had selected the Course:"

+ list1.getSelectedValue();

label.setText(data);

}

**if** (list2.getSelectedIndex() != -1) {

data += " and Hostel with the "

+ "facility of: ";

**for** (Object frame :

list2.~~getSelectedValues~~()) {

data += frame + " ";

}

}

label.setText(data);

}

});

// Reset the text fields

b2.addActionListener(

**new** ActionListener() {

**public** **void** actionPerformed(

ActionEvent e)

{

area2.setText("");

area1.setText(" ");

tf1.setText("");

tf2.setText("");

tf3.setText("");

tf4.setText("");

tf5.setText("");

tf6.setText(" ");

}

});

// Implementing the Print action

Print.addActionListener(

**new** ActionListener() {

**public** **void** actionPerformed(

ActionEvent e)

{

**try** {

area2.print();

}

**catch** (java.awt.print

.PrinterException a) {

System.***err***.format(

"NoPrinter Found",

a.getMessage());

}

}

});

// Generating the receipt

Receipt.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e)

{

area2.setText(

"--------------------------------"

+ "-----------Student Report----"

+ "--------------------------"

+ "--------------------------"

+ "-------------------\n");

area2.setText(area2.getText()

+ "Student Name: "

+ tf1.getText()

+ "\n");

area2.setText(area2.getText()

+ "Subject 1: "

+ tf3.getText()

+ "\n");

area2.setText(area2.getText()

+ "Subject 2: "

+ tf2.getText()

+ "\n");

area2.setText(area2.getText()

+ "Subject 3: "

+ tf4.getText()

+ "\n");

area2.setText(area2.getText()

+ "Contact Number: "

+ tf5.getText()

+ "\n");

area2.setText(area2.getText()

+ "Address: "

+ area1.getText()

+ "\n");

area2.setText(area2.getText()

+ "Gender: "

+ r5.getText()

+ "\n");

area2.setText(area2.getText()

+ "Nationality: "

+ tf6.getText()

+ "\n");

area2.setText(area2.getText()

+ "Pin code: "

+ tf7.getText()

+ "\n");

area2.setText(area2.getText()

+ "Branch: "

+ cb.getSelectedItem()

.toString()

+ "\n");

**if** (rb1.isSelected()) {

area2.setText(area2.getText()

+ "Student of "

+ "JSPM's ICOER "

+ "Wagholi\n");

}

**if** (rb2.isSelected()) {

area2.setText(area2.getText()

+ "Student of "

+ "BSIOTR "

+ "Wagholi\n");

}

**if** (rb3.isSelected()) {

area2.setText(area2.getText()

+ "Student with "

+ "Good marks \n");

}

**if** (rb4.isSelected()) {

area2.setText(area2.getText()

+ "Student needs "

+ "improvement \n");

}

area2.setText(area2.getText()

+ "Had chosen: "

+ list1.getSelectedValue()

.toString()

+ "\n");

area2.setText(area2.getText()

+ "Had chosen: "

+ list2.getSelectedValue()

.toString()

+ "\n");

**int** index2 = list2.getSelectedIndex();

**if** (index2 == 0) {

area2.setText(area2.getText()

+ " "

+ "Total amount to be "

+ "paid is 4 Lakhs \n");

}

**if** (index2 == 1) {

area2.setText(area2.getText()

+ " "

+ "Total amount to be paid "

+ "is 3.9 Lakhs \n");

}

**if** (index2 == 2) {

area2.setText(area2.getText()

+ " "

+ "Total amount to be paid "

+ "is 3.8 Lakhs \n");

}

**if** (index2 == 3) {

area2.setText(area2.getText()

+ " "

+ "Total amount to be paid "

+ "is 3.7 Lakhs \n");

}

**if** (index2 == 4) {

area2.setText(area2.getText()

+ " "

+ "Total amount to be paid "

+ "is 2.9 Lakhs \n");

}

**if** (e.getSource() == Receipt) {

**try** {

FileWriter fw

= **new** FileWriter(

"java.txt", **true**);

fw.write(area2.getText());

fw.close();

}

**catch** (Exception ae) {

System.***out***.println(ae);

}

}

JOptionPane.*showMessageDialog*(

area2, "DATA SAVED SUCCESSFULLY");

};

});

addWindowListener(

**new** WindowAdapter() {

**public** **void** windowClosing(

WindowEvent we)

{

System.*exit*(0);

}

});

setSize(800, 800);

setLayout(**null**);

setVisible(**true**);

setBackground(Color.***cyan***);

}

**public** **static** **void** main(String[] args)

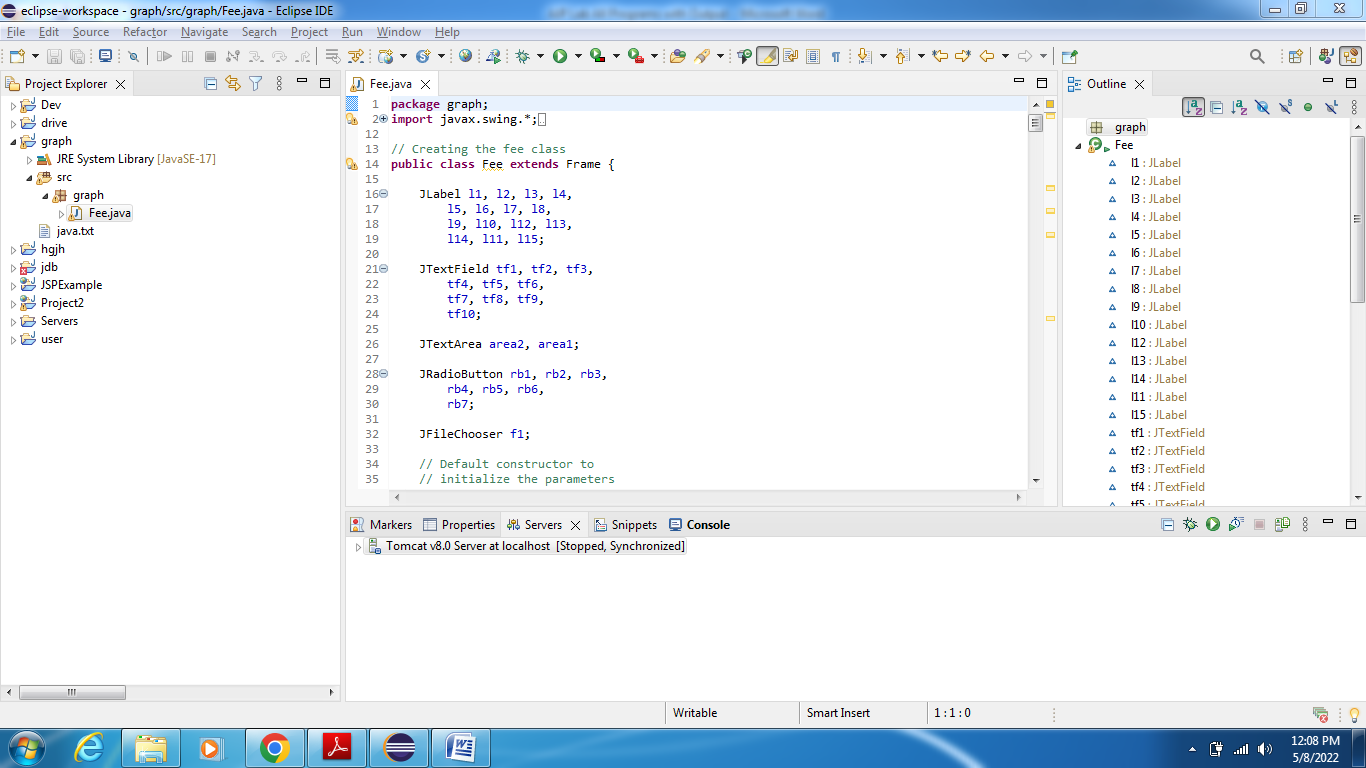
{

**new** Fee();

}

}

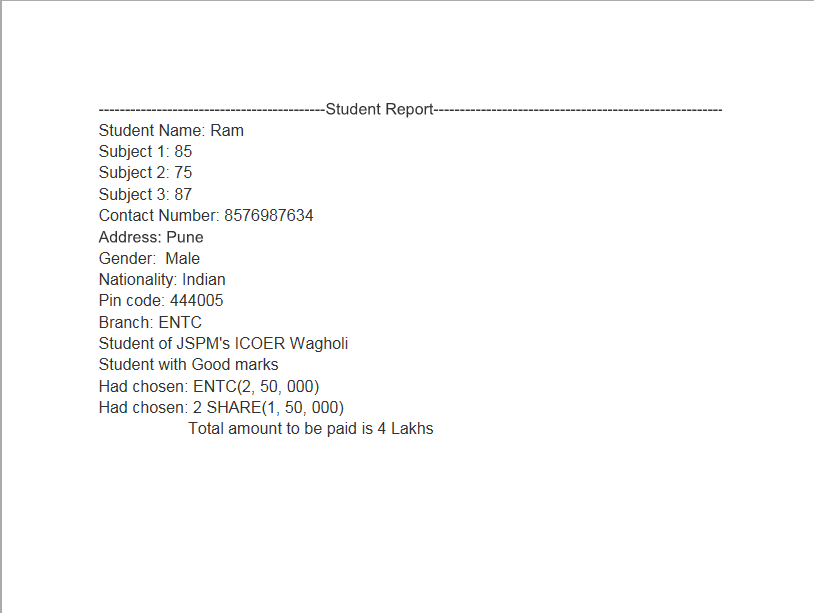
**Note –** Run in Eclipse.



**Output –**



**Report -**



\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Experiment No.: 4

# Name of Expt.: Write a program to insert and retrieve the data from the database using JDBC. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Program –**

**Conn.java**

**import** java.sql.\*;

**public** **class** Conn

{

**public** **static** **void** main(String args[])

{

**try**

{

Class.*forName*("com.mysql.jdbc.Driver");

Connection con=DriverManager.*getConnection*("jdbc:mysql://localhost:3306/db","root","root");

//here db is database name, root is username and root is password

Statement stmt=con.createStatement();

ResultSet rs=stmt.executeQuery("select \* from student");

**while**(rs.next())

{

**int** id=rs.getInt("id");

String city=rs.getString("city");

System.***out***.println(id+" "+city);

}

stmt.close();

con.close();

}**catch**(Exception e)

{

System.***out***.println(e);

}

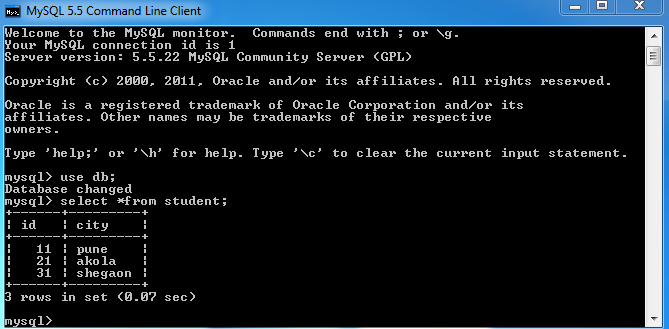
}

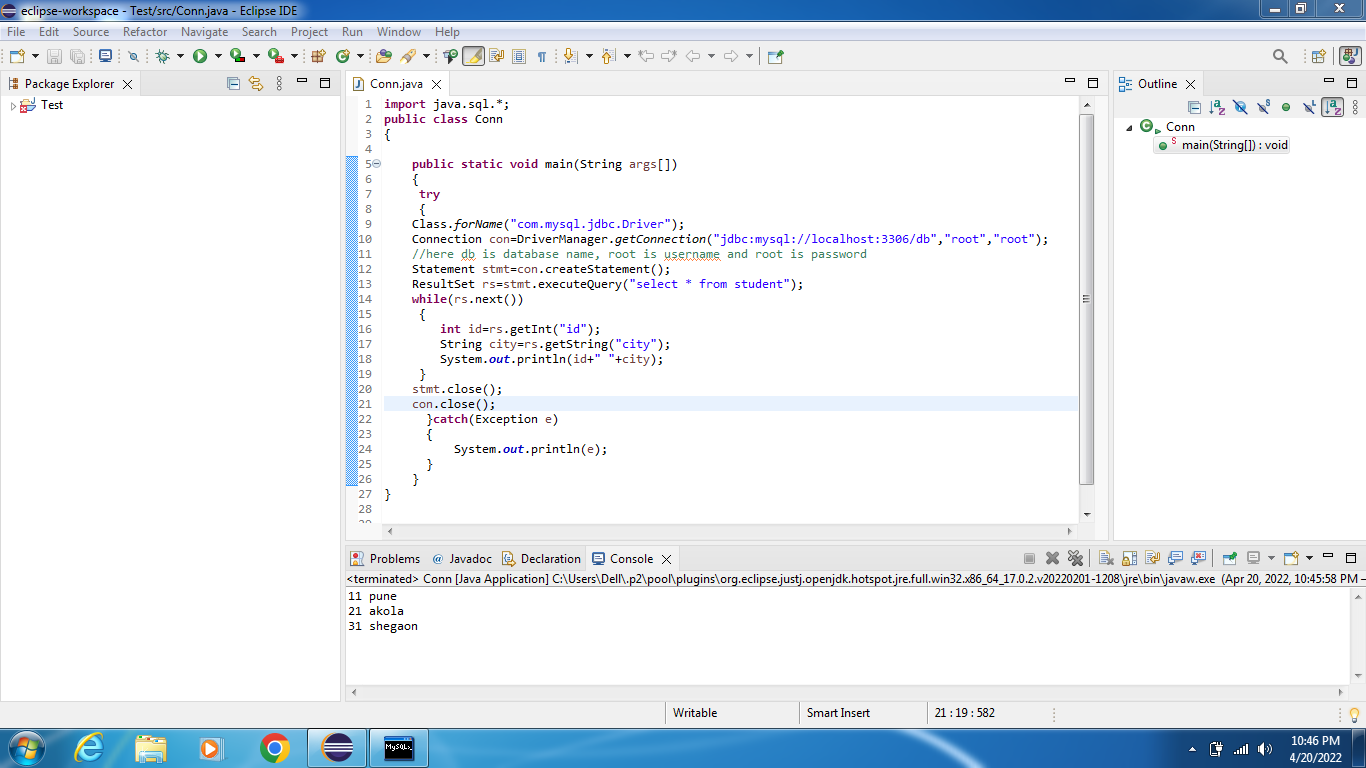
}

**Steps – Install MySQL in system.** (refer provided steps or from internet).

1. Open MySQL command promt
2. Put password as root
3. Create database – use appropriate command
4. Use database - use appropriate command
5. Create table - use appropriate command
6. Insert values in table - use appropriate command
7. Select \* from table - use appropriate command
8. Table will be displayed on MySQL command promt
9. Open eclipse.
10. Use above Conn.java file
11. Run above Conn.java file
12. Table will be displayed on console in eclipse.

**Output –**

****

****

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Experiment No.: 5

# Name of Expt.: Develop an RMI application which accepts a string or a number and checks that string or number is palindrome or not. \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Program –**

**One.java**

import java.rmi.\*;

interface one extends Remote

{

public int palin(String a) throws RemoteException;

}

**two.java**

import java.rmi.\*;

import java.lang.\*;

import java.rmi.server.\*;

public class two extends UnicastRemoteObject implements one

{

public two() throws RemoteException { }

public int palin(String a) throws RemoteException

{

System.out.println("Hello");

StringBuffer str = new StringBuffer(a);

String str1 = str.toString();

System.out.println("Print : " + str1.toString());

StringBuffer str2 = str.reverse();

System.out.println("Print : " + str2.toString());

int b = str1.compareTo(str2.toString());

System.out.println("Print : " + b);

if (b == 0)

return 1;

else

return 0;

}

}

**rmiserver.java**

import java.io.\*;

import java.rmi.\*;

import java.net.\*;

public class rmiserver

{

public static void main(String args[]) throws Exception

{

try

{

two twox = new two();

Naming.bind("palin", twox);

System.out.println("Object registered");

}

catch(Exception e)

{

System.out.println("Exception" + e);

}

}

}

**rmiclient.java**

import java.io.\*;

import java.rmi.\*;

import java.net.\*;

public class rmiclient

{

public static void main(String args[]) throws Exception

{

try

{

String s1 = "rmi://localhost/palin";

one onex = (one)Naming.lookup(s1);

int m = onex.palin("madam");

System.out.println("Print : " + m);

if (m == 1)

{

System.out.println("The given string is a Palindrome");

}

else

{

System.out.println("The given string is not a Palindrome");

}

}

catch (Exception e)

{

System.out.println("Exception" + e);

}

}

}

**Note –** 1**.** Open command promt two times and do as given in below photos.

2. Don’t use eclipse.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

# Experiment No.: 6

# Name of Expt.: Write a program to demonstrate the use of Inet address class and its factory methods.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Program –**

**Cplus.java**

**import** java.io.\*;

**import** java.net.\*;

**import** java.util.\*;

**class** cplus

{

**public** **static** **void** main(String args[])

**throws** UnknownHostException

{

// To get and print InetAddress of Local Host

InetAddress address1 = InetAddress.*getLocalHost*();

System.***out***.println("InetAddress of Local Host : "+ address1);

// To get and print InetAddress of Named Host

InetAddress address2 = InetAddress.*getByName*("45.22.30.39");

System.***out***.println("InetAddress of Named Host : "+ address2);

// To get and print ALL InetAddresses of Named Host

InetAddress address3[] = InetAddress.*getAllByName*("172.19.25.29");

**for** (**int** i = 0; i < address3.length; i++)

{

System.***out***.println("ALL InetAddresses of Named Host : "+ address3[i]);

}

// To get and print InetAddresses of

// Host with specified IP Address

**byte** IPAddress[] = { 125, 0, 0, 1 };

InetAddress address4 = InetAddress.*getByAddress*(IPAddress);

System.***out***.println("InetAddresses of Host with specified IP Address : "+ address4);

// To get and print InetAddresses of Host

// with specified IP Address and hostname

**byte**[] IPAddress2 = { 105, 22, (**byte**)223, (**byte**)186 };

InetAddress address5 = InetAddress.*getByAddress*("gfg.com", IPAddress2);

System.***out***.println("InetAddresses of Host with specified IP Address & hostname :"+ address5);

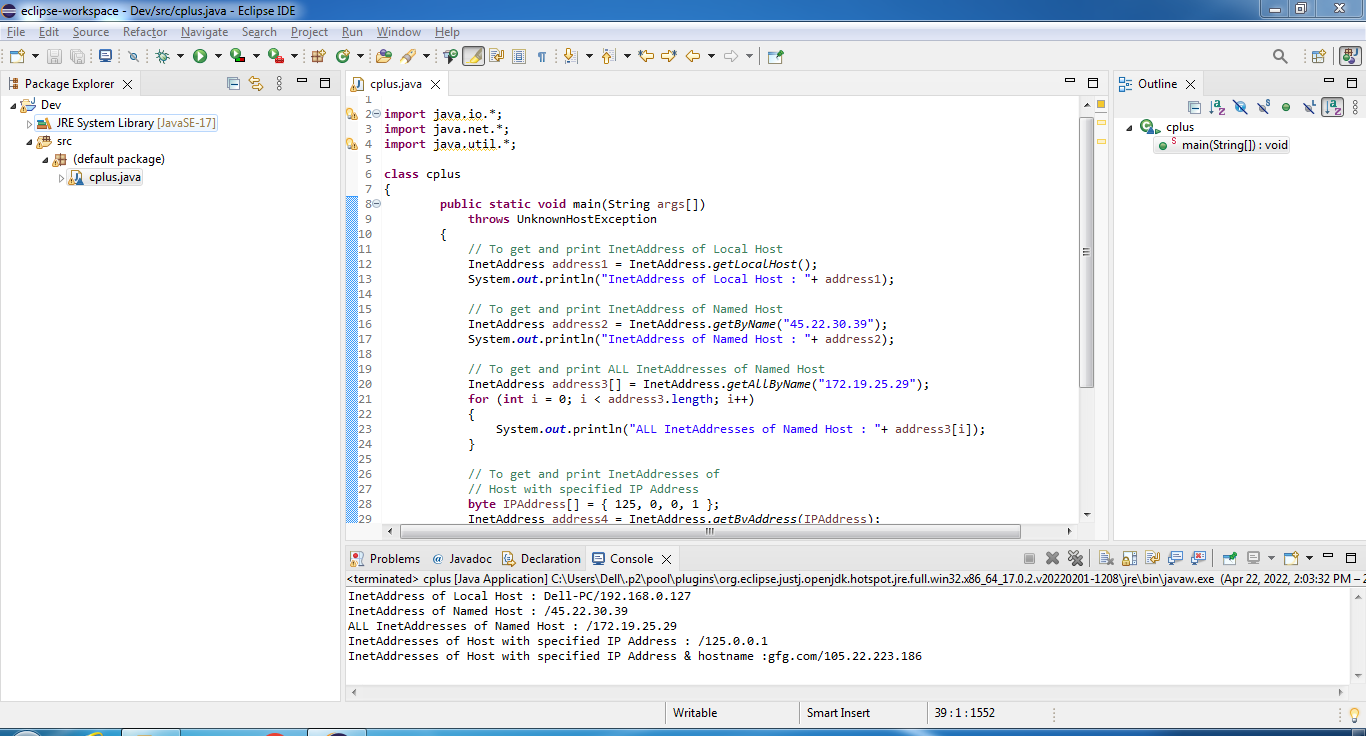
}

}

**Steps –**

1. Open Eclispe.
2. Copy above cplus.java file.
3. Run the cplus.java file.
4. Output will be displayed on console in Eclispe.

**Output -**

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

# Experiment No.: 7

**Name of Expt.: Write program with suitable example to develop your remote interface, implement your RMI server, implement application that create your server, also develop security policy file.**

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Program –**

**Hello.java**

import java.rmi.Remote;

import java.rmi.RemoteException;

// Creating Remote interface for our application

public interface Hello extends Remote {

void printMsg() throws RemoteException;

}

**ImplExample.java**

// Implementing the remote interface

public class ImplExample implements Hello {

// Implementing the interface method

public void printMsg() {

System.out.println("This is an example RMI program");

}

}

**Server.java**

import java.rmi.registry.Registry;

import java.rmi.registry.LocateRegistry;

import java.rmi.RemoteException;

import java.rmi.server.UnicastRemoteObject;

public class Server extends ImplExample {

public Server() {}

public static void main(String args[]) {

try {

// Instantiating the implementation class

ImplExample obj = new ImplExample();

// Exporting the object of implementation class

// (here we are exporting the remote object to the stub)

Hello stub = (Hello) UnicastRemoteObject.exportObject(obj, 0);

// Binding the remote object (stub) in the registry

Registry registry = LocateRegistry.getRegistry();

registry.bind("Hello", stub);

System.err.println("Server ready");

} catch (Exception e) {

System.err.println("Server exception: " + e.toString());

e.printStackTrace();

}

}

}

**Client.java**

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

public class Client {

private Client() {}

public static void main(String[] args) {

try {

// Getting the registry

Registry registry = LocateRegistry.getRegistry(null);

// Looking up the registry for the remote object

Hello stub = (Hello) registry.lookup("Hello");

// Calling the remote method using the obtained object

stub.printMsg();

// System.out.println("Remote method invoked");

} catch (Exception e) {

System.err.println("Client exception: " + e.toString());

e.printStackTrace();

}

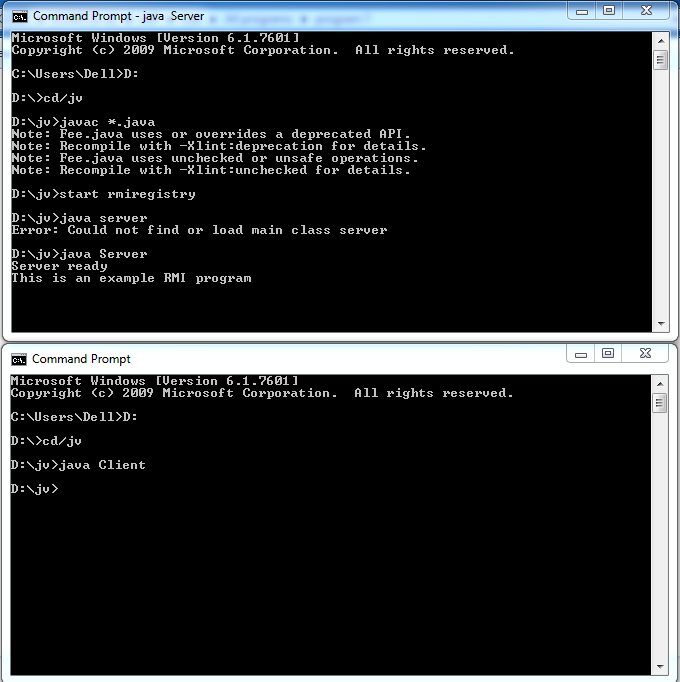
}

}

**Steps –**

1. Open command promt two times and do as given in below photos.
2. Don’t use eclipse.

**Output –**

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

# Experiment No.: 8

**Name of Expt.: Write** **a database application that uses any JDBC driver.**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

# Program –

# Conn.java -

**import** java.sql.\*;

**public** **class** Conn

{

**public** **static** **void** main(String args[])

{

**try**

{

Class.*forName*("com.mysql.jdbc.Driver");

Connection con=DriverManager.*getConnection*("jdbc:mysql://localhost:3306/db","root","root");

//here db is database name, root is username and root is password

Statement stmt=con.createStatement();

ResultSet rs=stmt.executeQuery("select \* from student");

**while**(rs.next())

{

**int** id=rs.getInt("id");

String city=rs.getString("city");

System.***out***.println(id+" "+city);

}

stmt.close();

con.close();

}**catch**(Exception e)

{

System.***out***.println(e);

}

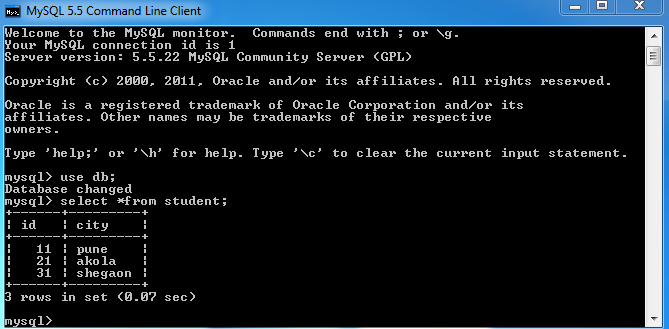
}

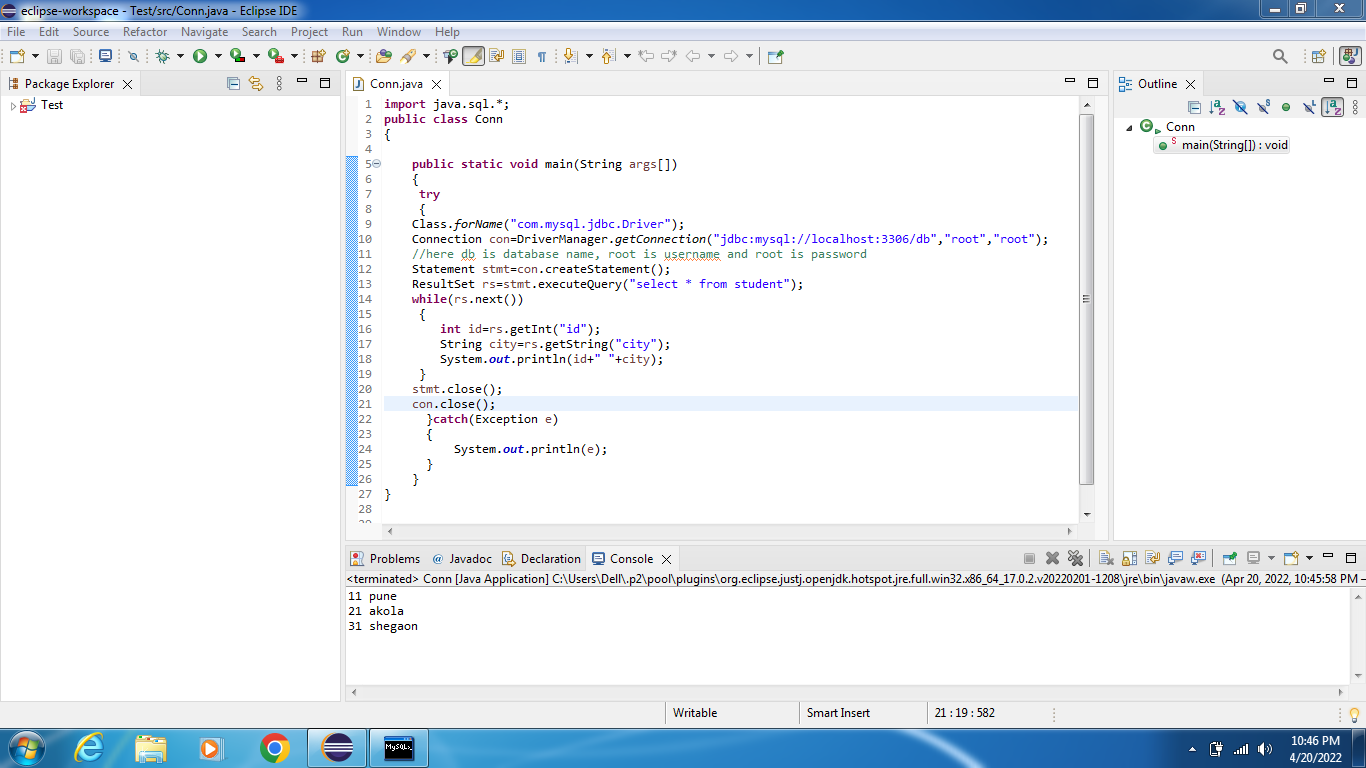
}

**Steps – Install MySQL in system.** (refer provided steps or from internet).

1. Open MySQL command promt
2. Put password as root
3. Create database – use appropriate command
4. Use database - use appropriate command
5. Create table - use appropriate command
6. Insert values in table - use appropriate command
7. Select \* from table - use appropriate command
8. Table will be displayed on MySQL command promt
9. Open eclipse.
10. Use above Conn.java file
11. Run above Conn.java file
12. Table will be displayed on console in eclipse.

**Output –**

****

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

# Experiment No.: 9

**Name of Expt.: Write** **a simple JSP page to display a simple message (It may be a simple html page).**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Program –**

**myFirstJSP.jsp –**

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"

"http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>First JSP page</title>

</head>

<body>

<center>

<font color="blue" size="10"> Hello World! This is first JSP page </font>

</center>

</body>

</html>

**Steps –** Configure Apache Tomcat in eclipse.(Apache tomcat should be already available on system)

1. Open eclipse.
2. Click on help tab.
3. Click on install new software.
4. Follow remaining steps to update eclipse.
5. Now go to new tab in eclipse.
6. Create dynamic web project (now pop up window will open on which browse option is there to browse Apache Tomcat.)

**Steps –** 1. Go to new tab in eclipse.

2**.** Create dynamic web project.

3. Create JSP program(refer above myFirstJSP.jsp program)**.**

4. Right click on above jsp program and click on Run as – run on server.

5. Check output on internet explorer browser which is html page.

**Output –**



**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

# Experiment No.: 10

**Name of Expt.: Write Create a simple calculator application using servlet.**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Program –**

**Calculator.java**

package Perform;

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

public class Calculator extends HttpServlet

{

public void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException

{

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

out.println("<html><head><title>Calculator</title></head><body>");

double n1 = Double.parseDouble(request.getParameter("txtN1"));

double n2 = Double.parseDouble(request.getParameter("txtN2"));

double result =0;

String opr=request.getParameter("opr");

if(opr.equals("+")) result=n1+n2;

if(opr.equals("-")) result=n1-n2;

if(opr.equals("\*")) result=n1\*n2;

if(opr.equals("/")) result=n1/n2;

out.println("<h1> Result = "+result);

out.println("</body></html>");

}

}

**Index.html –**

<!DOCTYPE html>

<html>

<head>

<title>Calculator App</title>

</head>

<body>

<form action="CalculatorServlet" method="post" >

Enter First Number <input type="text" name="txtN1" ><br>

Enter Second Number <input type="text" name="txtN2" ><br>

Select an Operation

<input type="radio" name="opr" value="+">ADDTION

<input type="radio" name="opr" value="\*">MULTIPLY

<input type="radio" name="opr" value="/">DIVIDE

<input type="radio" name="opr" value="-">SUB

<br> <input type="reset">

<input type="submit" value="Calculate" >

</form>

</body>

</html>

**Web.xml –**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"http://xmlns.jcp.org/xml/ns/javaee"* xsi:schemaLocation=*"http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app\_3\_1.xsd"* id=*"WebApp\_ID"* version=*"3.1"*>

<display-name>Project2</display-name>

<welcome-file-list>

<welcome-file>index.html</welcome-file>

<welcome-file>index.jsp</welcome-file>

<welcome-file>index.htm</welcome-file>

<welcome-file>default.html</welcome-file>

<welcome-file>default.jsp</welcome-file>

<welcome-file>default.htm</welcome-file>

</welcome-file-list>

<servlet>

<servlet-name>Calculator</servlet-name>

<servlet-class>Calculator</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>Calculator</servlet-name>

<url-pattern>Calculator</url-pattern>

</servlet-mapping>

</web-app>

**Steps –** Configure Apache Tomcat in eclipse.(Apache tomcat should be already available on system)

1. Open eclipse.
2. Click on help tab.
3. Click on install new software.
4. Follow remaining steps to update eclipse.
5. Now go to new tab in eclipse.
6. Create dynamic web project (now pop up window will open on which browse option is there to browse Apache Tomcat.)

**Steps –** 1. Go to new tab in eclipse.

2**.** Create dynamic web project.(provide project name).

3. Right click on project name(provided name) in package explorer of eclipse.

4. Create html program (refer above index.html program).

5. Right click on project name(provided name) in package explorer of eclipse.

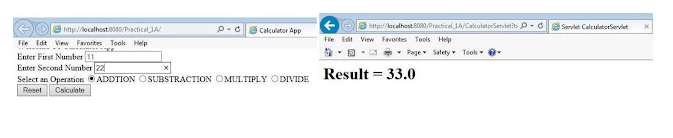
6. Create servlet program(refer above calculator.java program)**.**

7. Click on project name(provided name) in package explorer of eclipse, go to web.xml file in src folder.(refer above web.xml program)

8. Right click on html program and click on Run as – run on server.

9. Check output on internet explorer browser which is html page.

**Output-**

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*ALL THE BEST\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***